



SyScan^{'14}

RFIDLER

a software defined RFID tool

Adam Laurie
(Zac Franken)

Who are we?

- Aperture labs:
www.aperturelabs.com

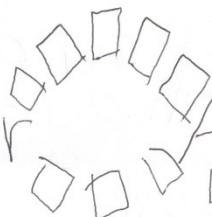


Aperture Labs



Fan Mail

April 1, 2013

Dear  Aperture
Laboratories

Do you make ~~the~~
Portal Guns?

Do they work?
well I have a idea
for a portal Gun.
here is the picture 

The portal
colors are
Yellow and
Rain bow

From
Joshua



Who are we?

- Zac Franken
 - Chip Monkey
 - Scary Chemicals
 - Bad Smells



Who are we?

- Adam Laurie
 - Code Monkey
 - Convert scary analogue Magic Moonbeams to lovely Digital Bits & Bytes





POC @dc4420
<http://dc4420.org>

@rfidiot

<http://adamsblog.aperturelabs.com>

What?

- RFIDler
 - Software Defined RFID

Why?

- Many systems totally insecure
 - Manufacturers know it
- Existing tools expensive / complicated
 - Proxmox3
 - Very good but 'fragile' & expensive
- Vendor specific dev kits
 - Locked in to one tag type
- Disrupt the market
 - Change threat landscape

Why?

- RFID is confusing
 - Proliferation of standards
 - Proprietary systems
 - Analogue
 - Inductive Coupling / NFC
 - Magic Moonbeams
 - Digital only after **ALL** decoding/demodulation

Software Defined Radio

- RF front-end in hardware
- Everything else in software
 - Modulation
 - Filtering
 - Mixing
 - etc.

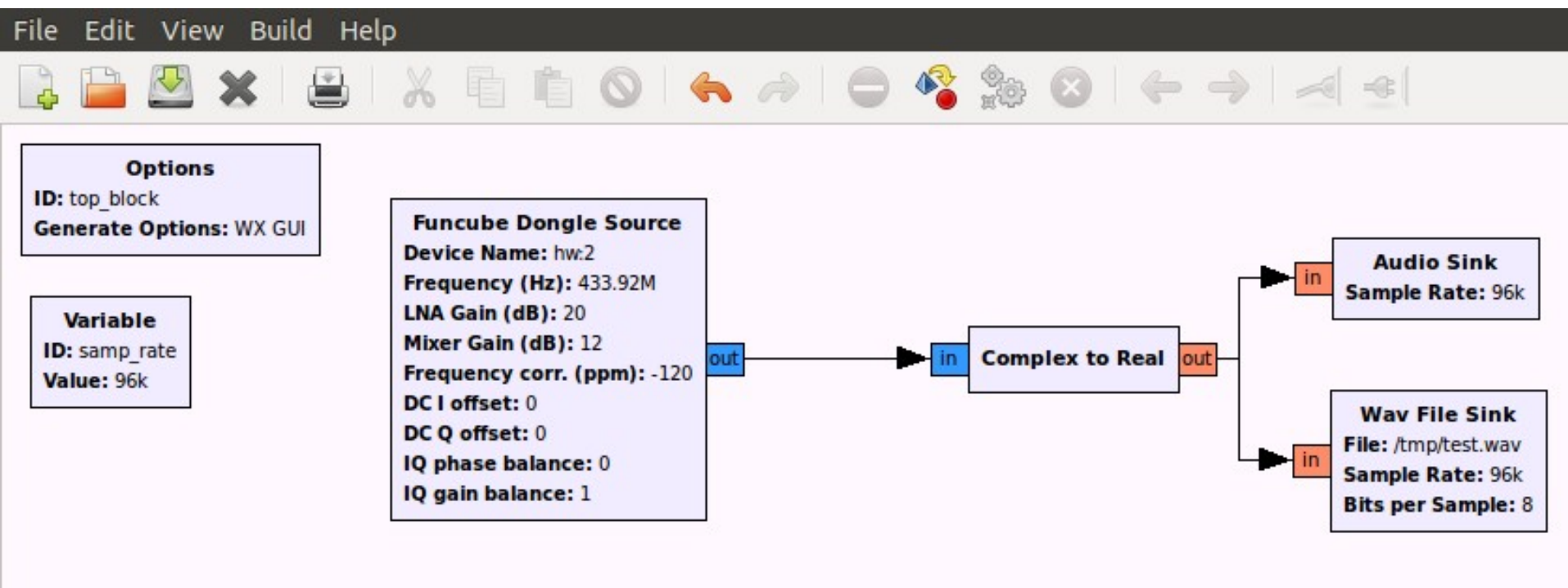
Software Defined Radio

- FUNcube Dongle
 - 150 kHz -> 1.9GHz



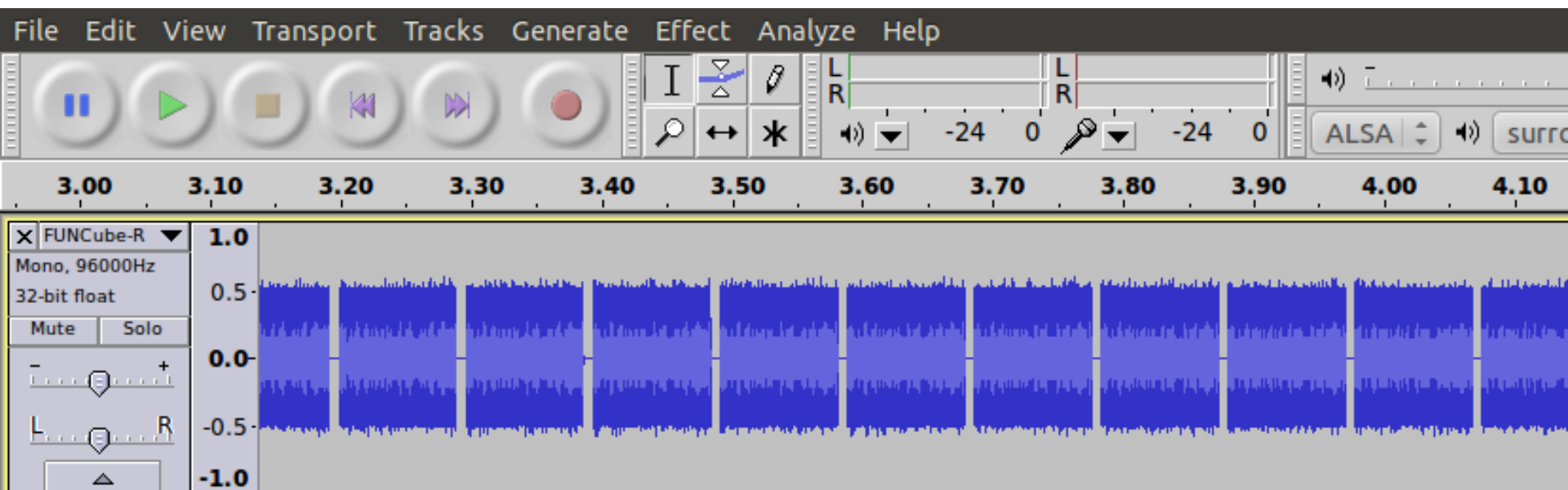
Software Defined Radio

- GNU Radio Companion
 - Works with any supported hardware
 - Creates python code to drive GNU Radio



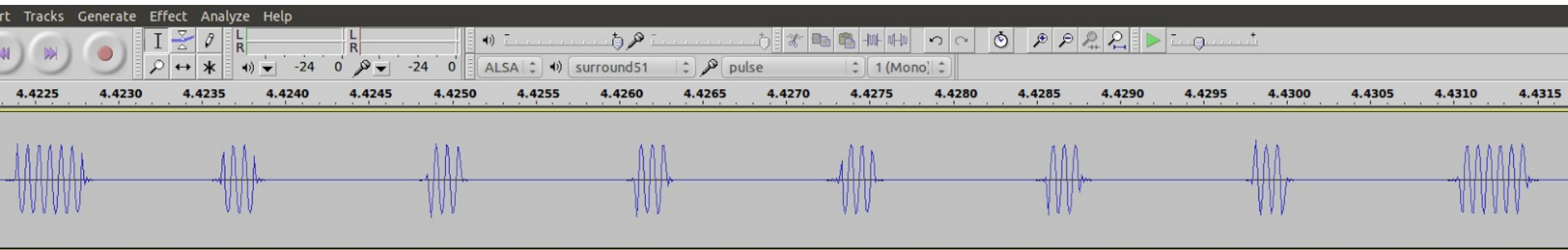
Software Defined Radio

- Raw data capture
 - Saved as WAV file



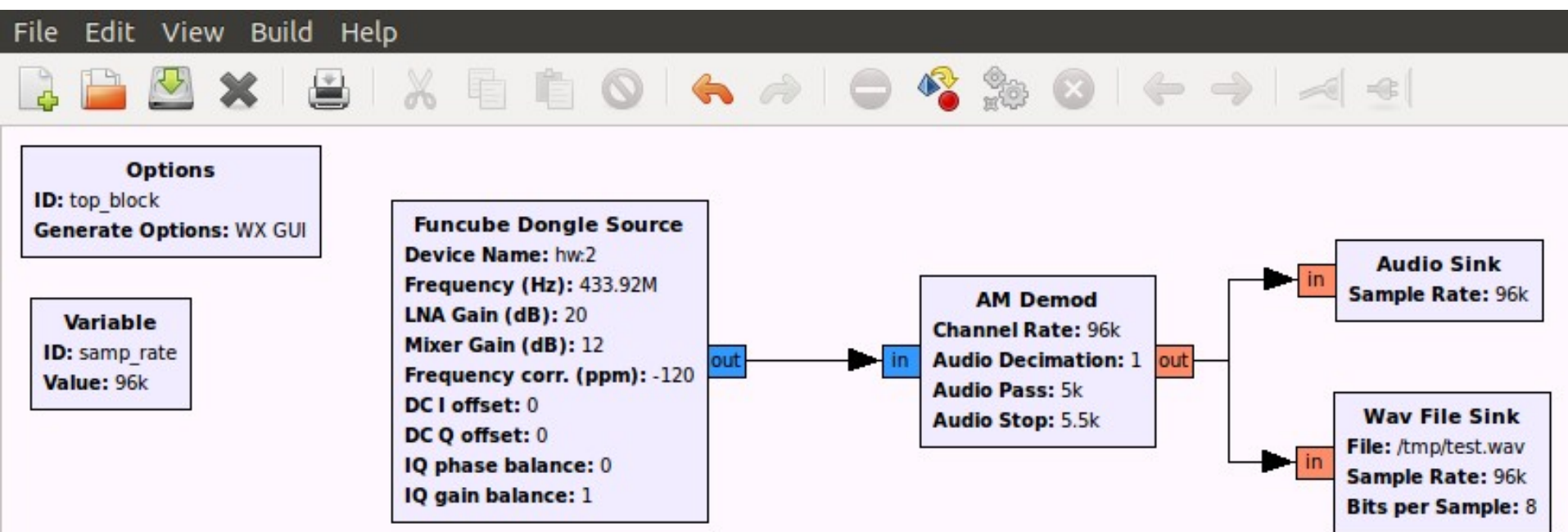
Software Defined Radio

- Raw data
 - AM - Amplitude Modulation
 - OOK – On / Off Keying



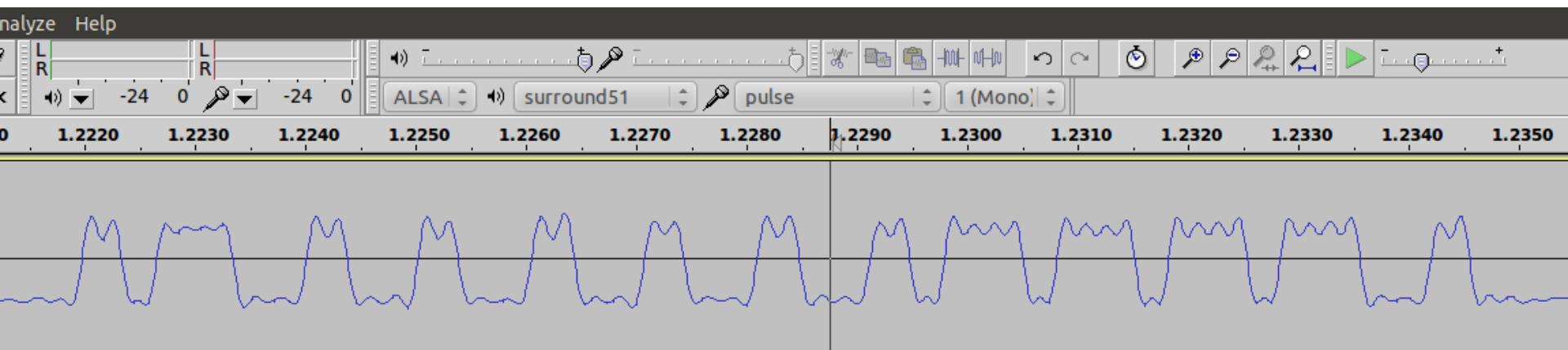
Software Defined Radio

- GNU Radio Companion
 - Pre-defined modulators / de-modulators
 - AM



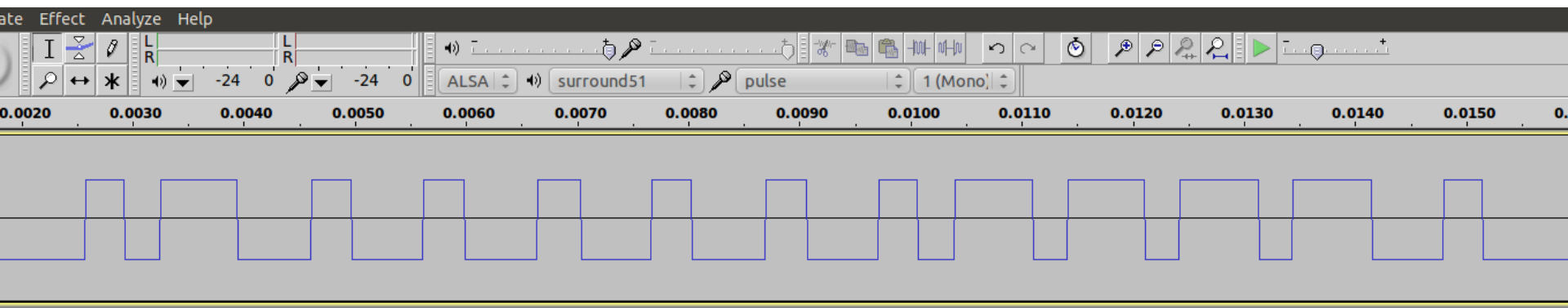
Software Defined Radio

- AM data capture
 - Saved as WAV file



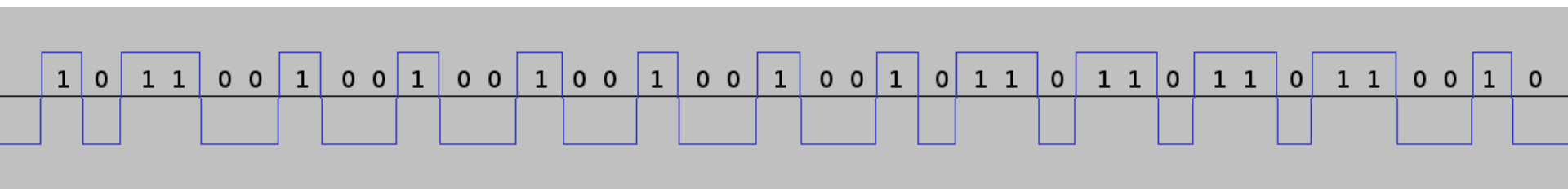
Software Defined Radio

- AM data capture
 - Convert to square wave



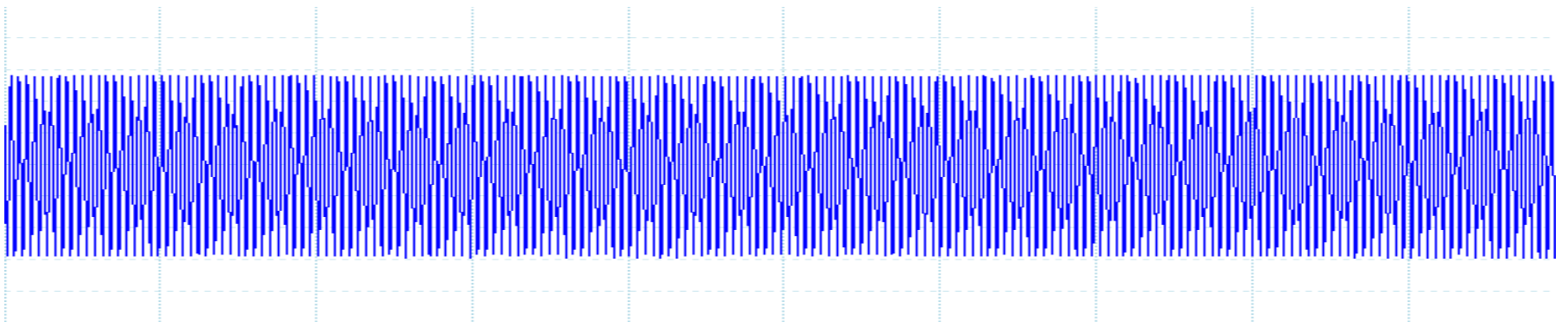
Software Defined Radio

- Decode to binary
 - HIGH is 1
 - LOW is 0
 - Smallest pulse is single bit length
 - 10110010010010010010010110110110110010



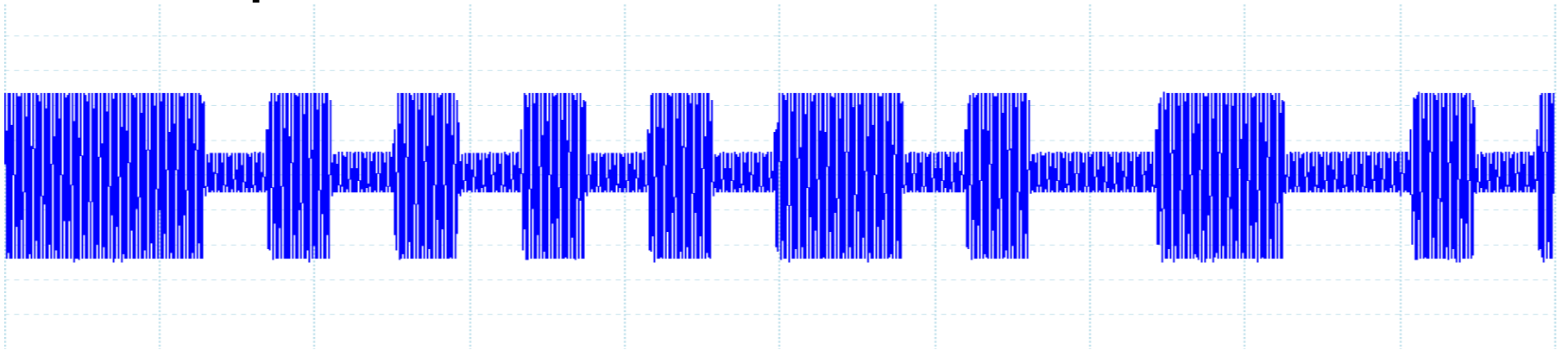
RFID Basics

- TAG and READER are inductively coupled
- READER generates CARRIER (in this case 125KHz) to energise TAG
- TAG takes power from its coil



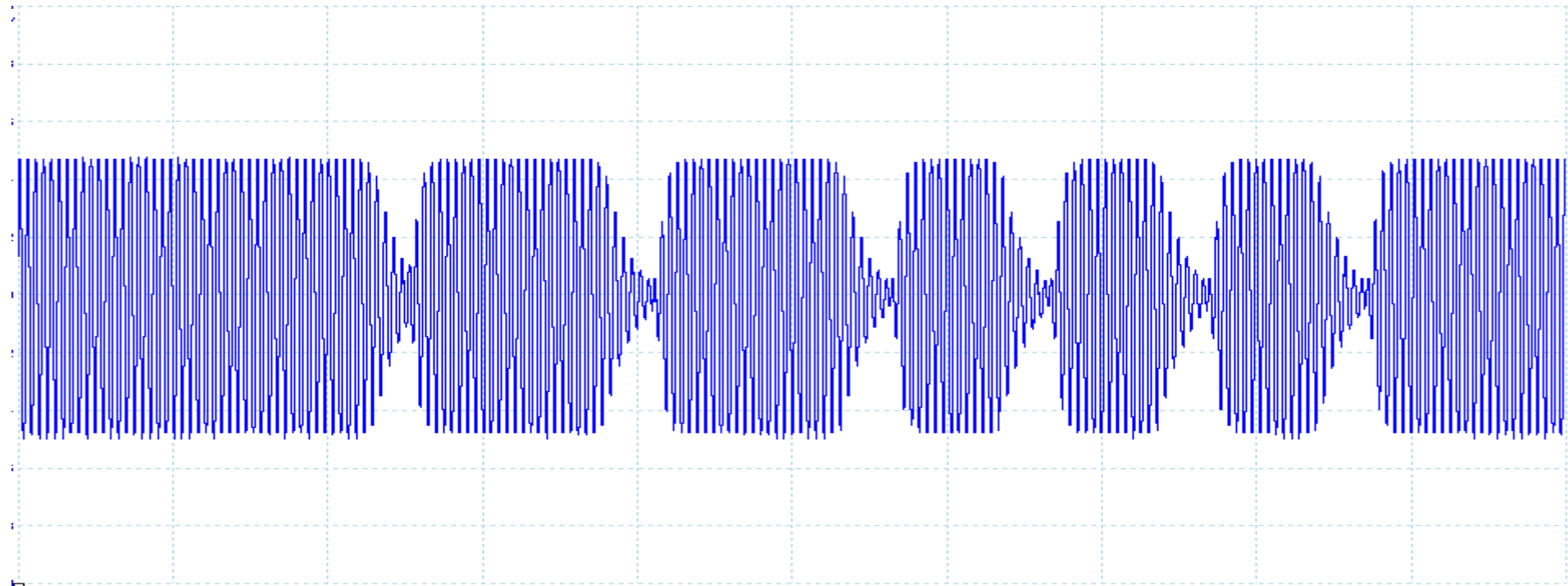
RFID Basics

- TAG communicates to READER by grounding its coil, thereby inducing a voltage drop in the inductively coupled READER coil



RFID Basics

- Reader communicates to TAG by interrupting the CARRIER



RFID Basics

- Modulation:
 - ASK – Amplitude Shift Keying
 - OOK – On / Off Keying

RFID Basics

- Modulation:
 - ASK – Amplitude Shift Keying
 - OOK – On / Off Keying
 - READER ENERGISING coil
 - 'ON'
 - or not 'OFF'

RFID Basics

- Modulation:
 - ASK – Amplitude Shift Keying
 - OOK – On / Off Keying
 - READER ENERGISING coil
 - 'ON'
 - or not 'OFF'
 - TAG GROUNDING coil
 - 'ON'
 - or not 'OFF' (DAMPING)

RFID Basics

- Modulation **schemes**
 - ASK – Amplitude Shift Keying
 - OOK – On / Off Keying

RFID Basics

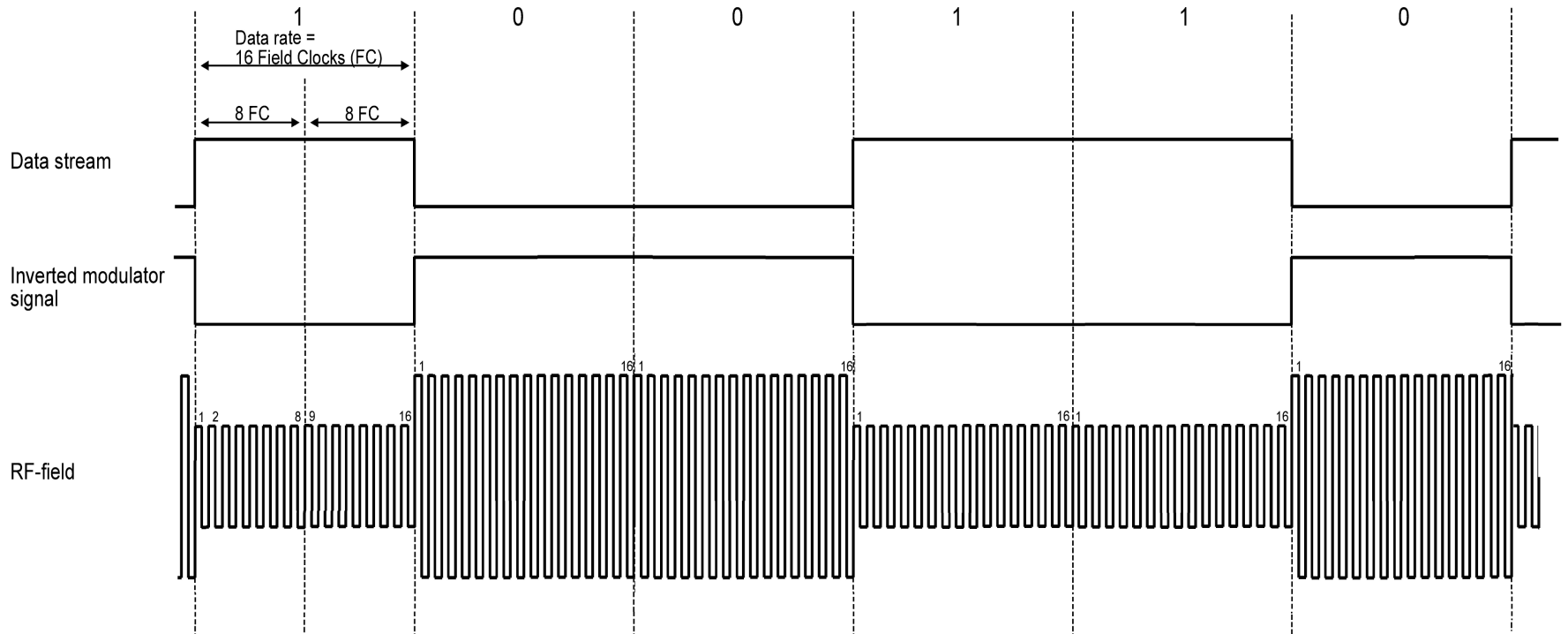
- Modulation **schemes**
 - ASK – Amplitude Shift Keying
 - OOK – On / Off Keying
 - That's all she wrote!

RFID Basics

- Modulation **schemes**
 - ASK – Amplitude Shift Keying
 - OOK – On / Off Keying
 - PWM – Pulse Width Modulation
 - FSK – Frequency Shift Keying
 - PSK – Phase Shift Keying
 - Manchester / BiPhase Encoding

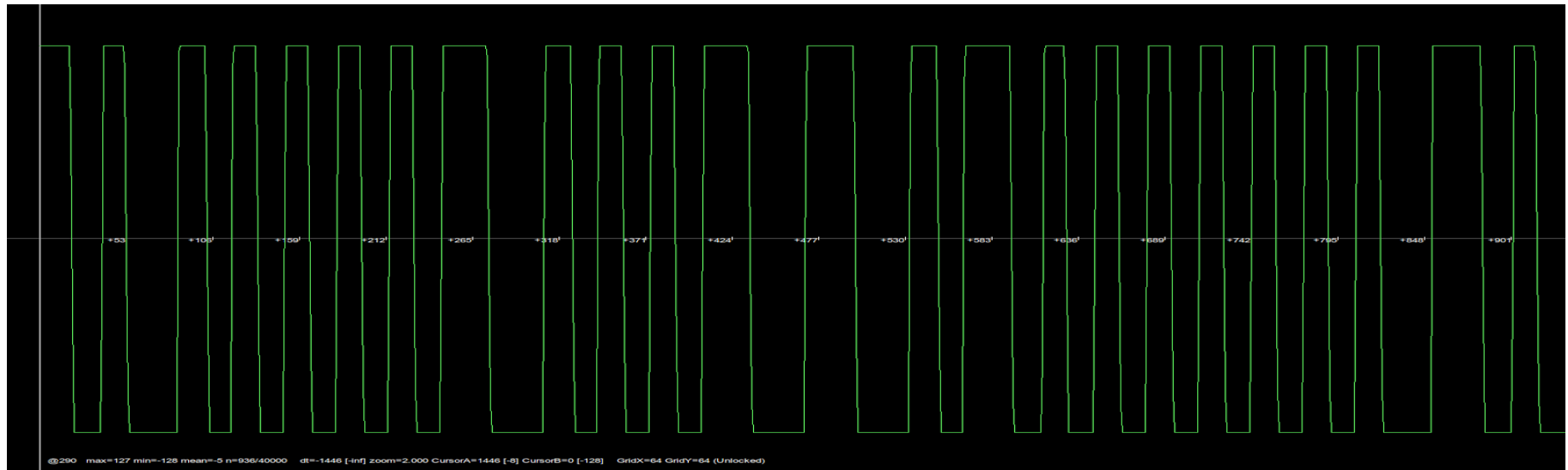
RFID Basics

- ASK / OOK



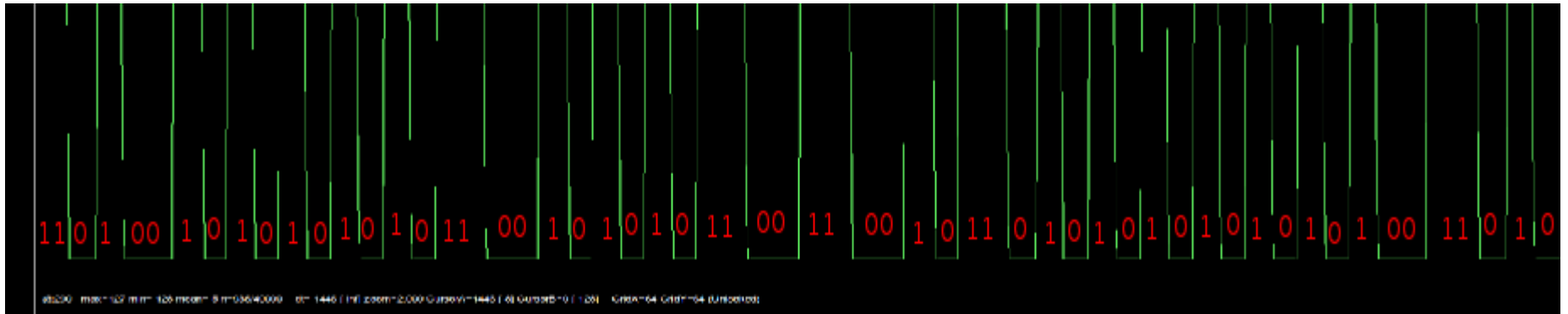
RFID Basics

- ASK / OOK
 - DAMPED for a 0
 - UN-DAMPED for a 1



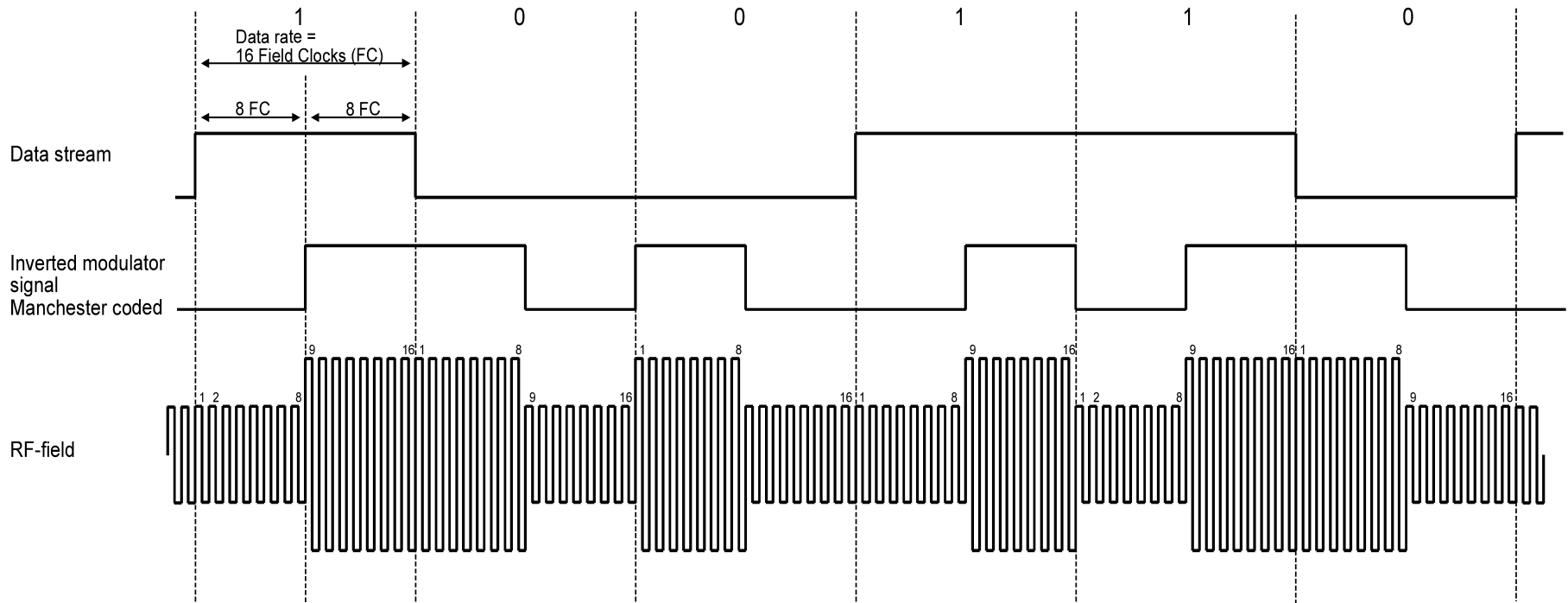
RFID Basics

- ASK – Amplitude Shift Keying
 - OOK – On / Off Keying
 - 1101001010101011001010101100110010110101010101010011010



RFID Basics

- Manchester encoding



RFID Basics

- Manchester encoding:
 - 11010010101010110010101011
0011001011010101010101010011
010
 - 10 = '1'
 - 01 = '0'
 - 11 = Invalid!
 - 00 = Invalid!

RFID Basics

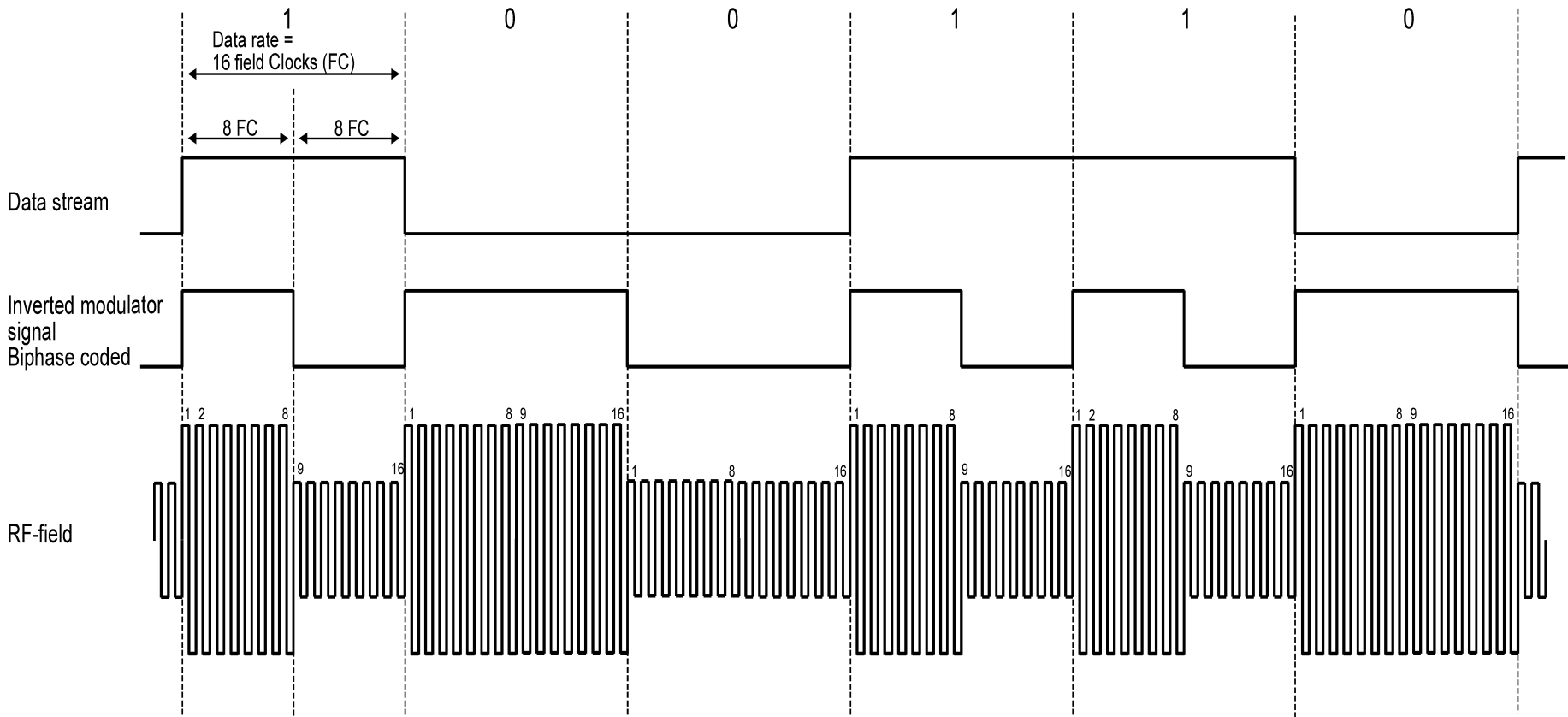
- Manchester encoding:
 - Two baseband pulse widths
 - '00' or '11' = long
 - '01' or '10' = 2 x short
 - Two pulse periods per bit
 - 11 01 00 10 10 10 10 10 11 00 10 10 10 10 11 00 11 00 10 11 01 01 01 01 01 01 0 10 10 01 10 10
 - Automatic error detection
 - 11 == Invalid!
 - Self clocking
 - Skip ½ bit:
 - 10 10 01 01 01 01 01 01 10 01 01 01 01 10 01 10 01 01 10 10 10 10 10 10 10 10 01 10 10
 - 11000000100001010011111111011

RFID Basics

- Manchester encoding:
 - Self-Clocking
 - Error-Detection
 - Ability to transmit ASK '0'
 - Distinguish from silence

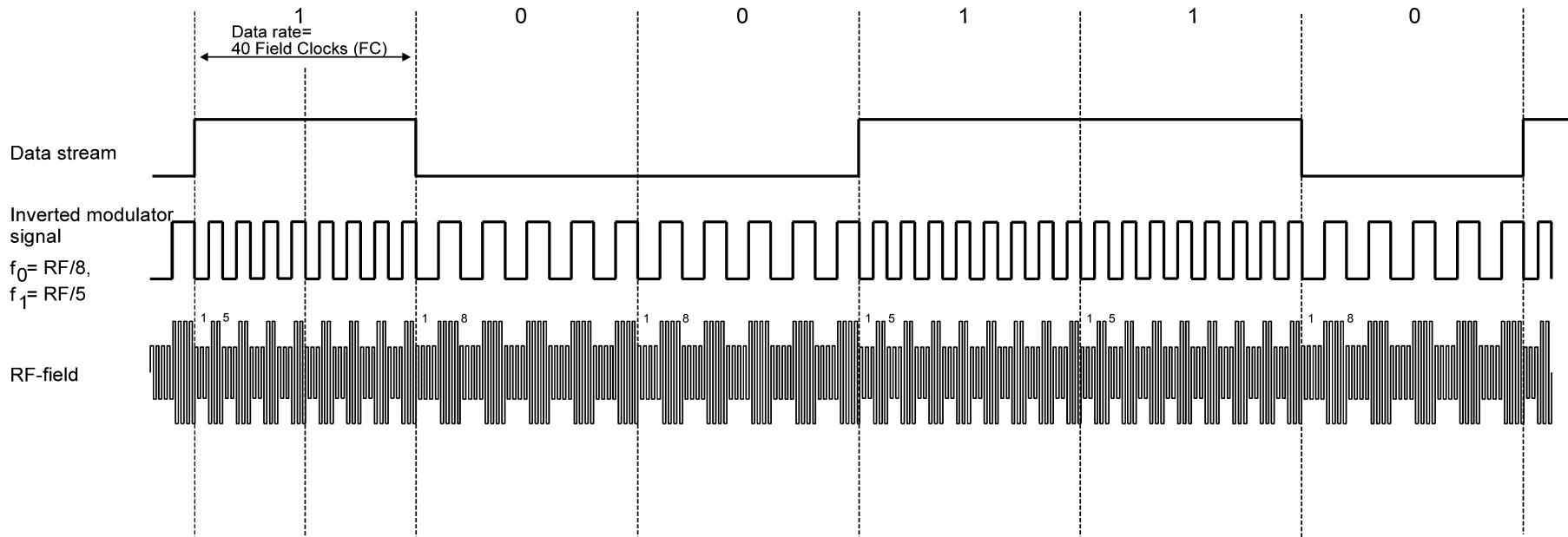
RFID Basics

- BiPhase encoding



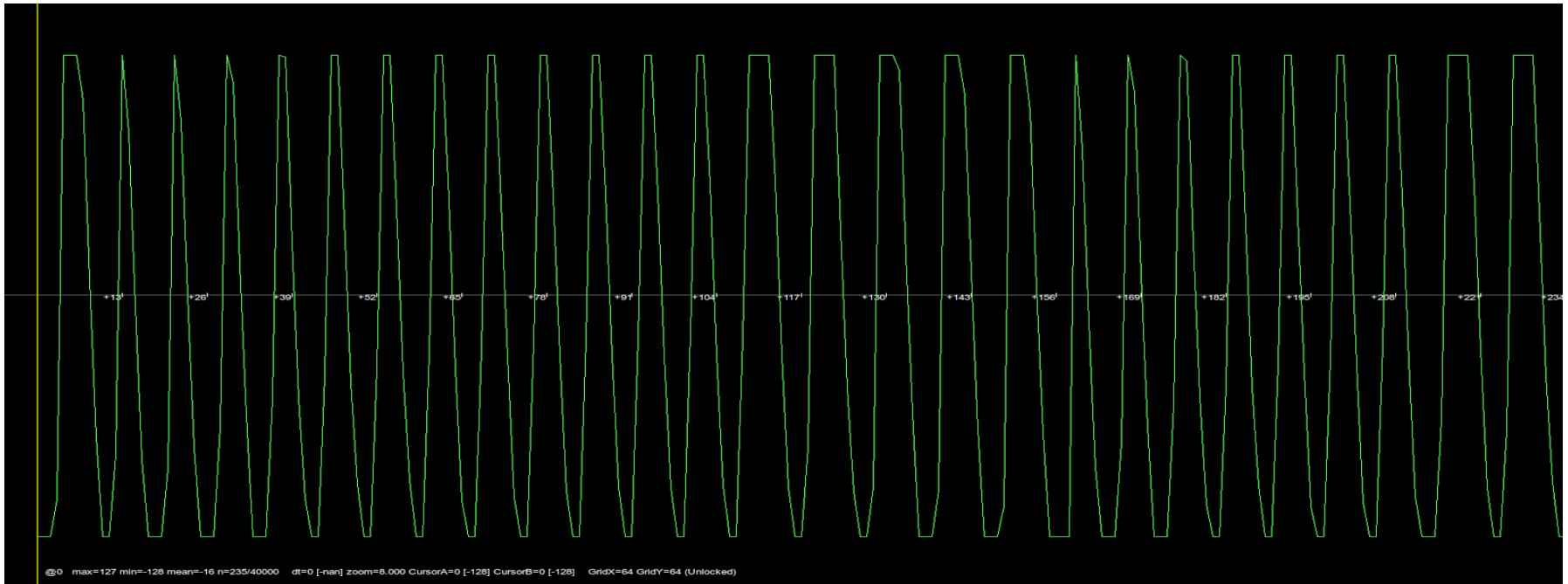
RFID Basics

- Modulation schemes
 - FSK – Frequency Shift Keying



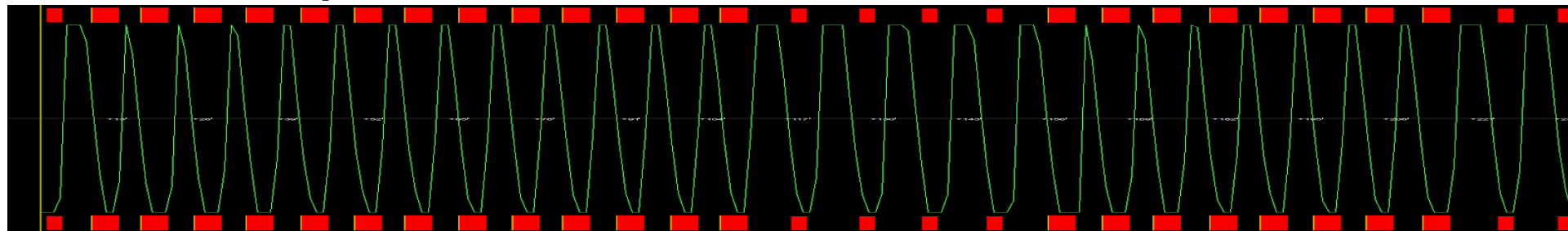
RFID Basics

- Modulation schemes
 - FSK – Frequency Shift Keying



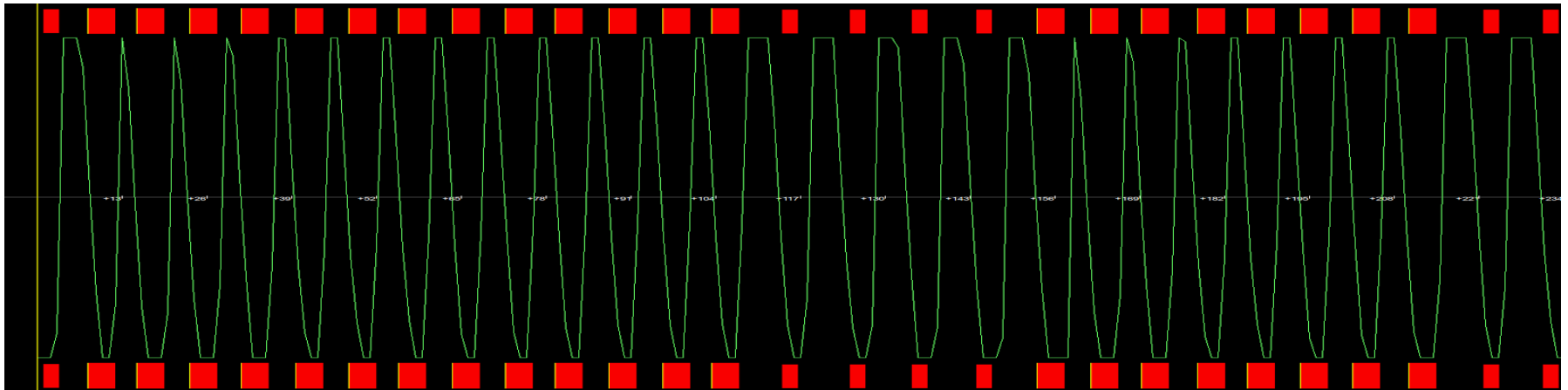
RFID Basics

- ASK / FSK – Frequency Shift Keying
 - DAMPING creates secondary pulses by allowing bursts of carrier
 - Frequency of pulses over fixed period determines '0' or '1'



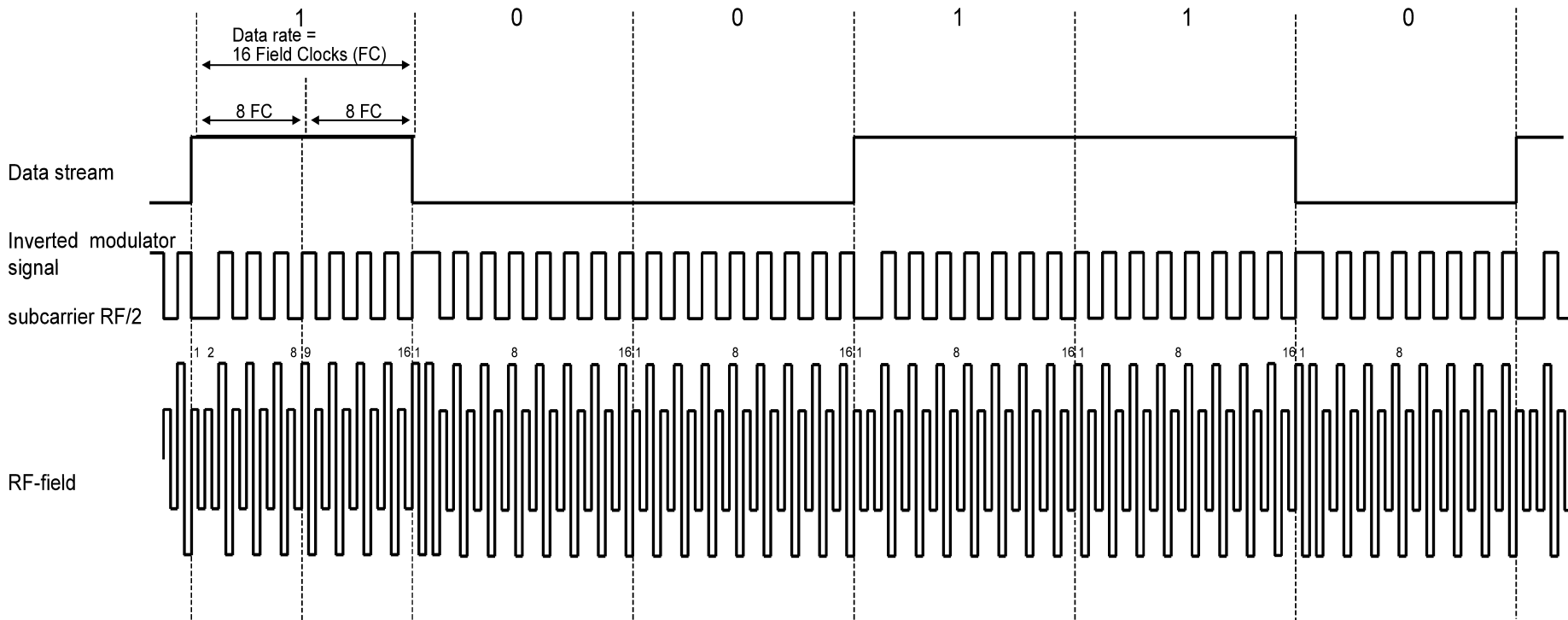
RFID Basics

- ASK / FSK – Frequency Shift Keying
 - 6 short = '0'
 - 5 long = '1'
 - This message: 100101



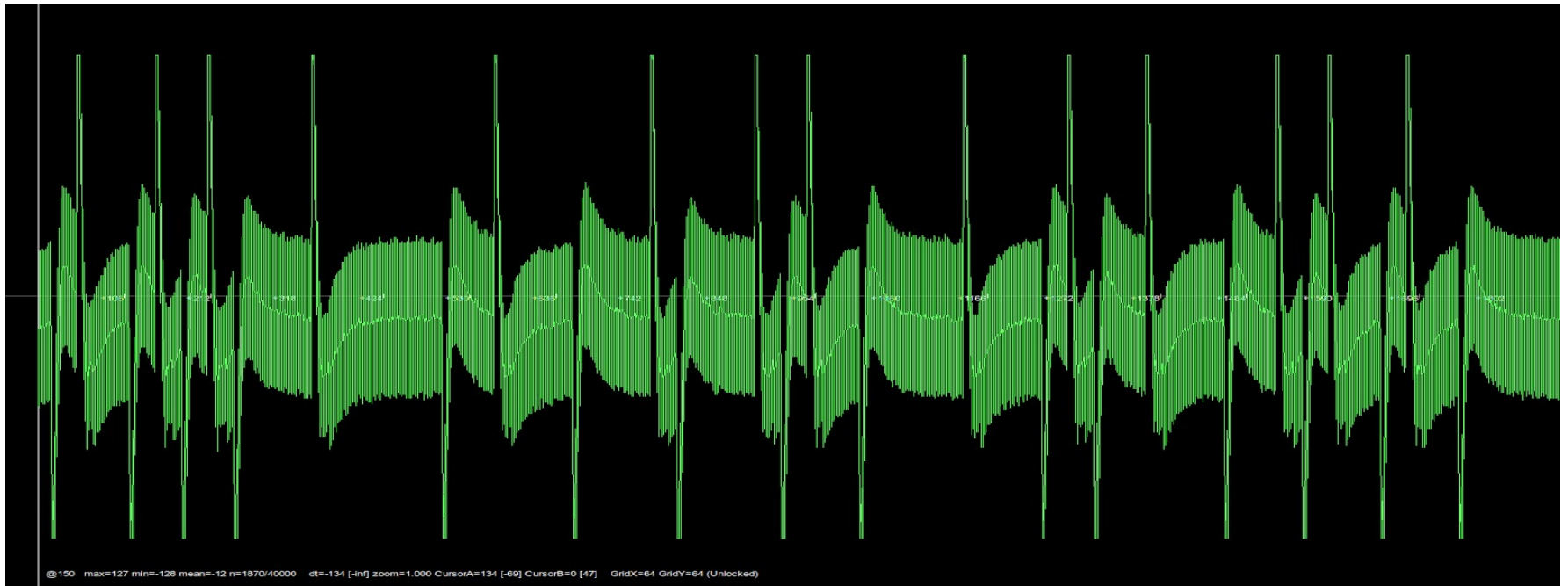
RFID Basics

- Modulation schemes
 - PSK – Phase Shift Keying



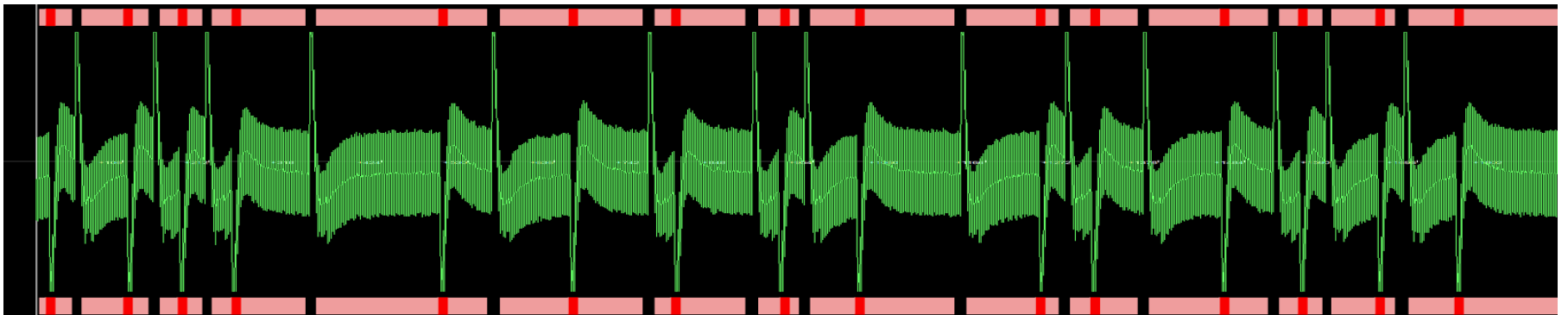
RFID Basics

- Modulation schemes
 - PSK – Phase Shift Keying



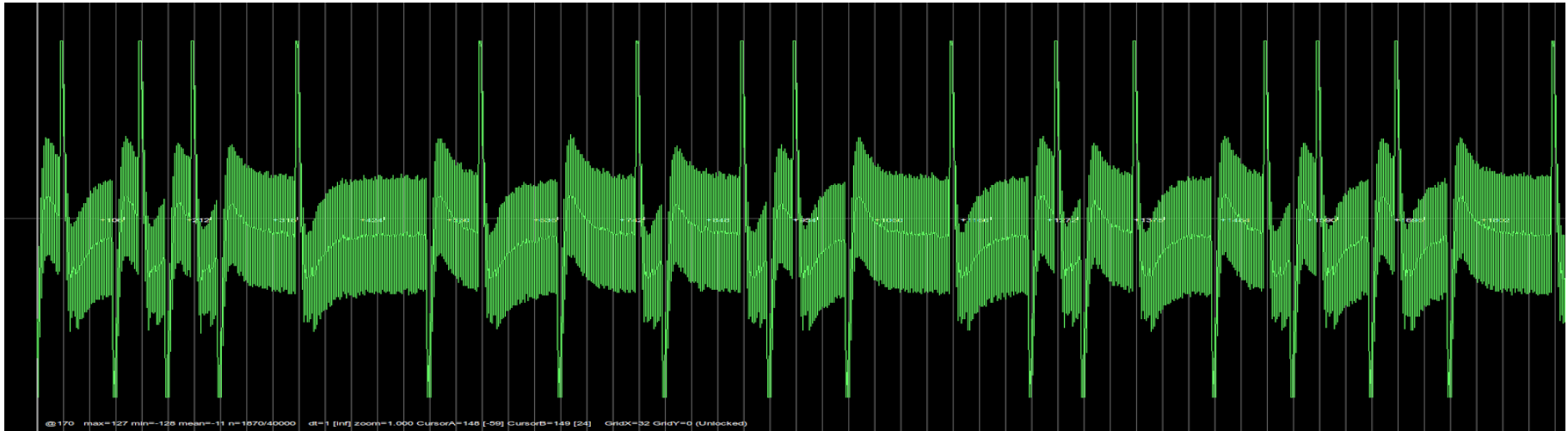
RFID Basics

- ASK / PSK – Phase Shift Keying
 - 50% DAMPING creates secondary CARRIER
 - Phase shift allows single burst of original CARRIER to break through
 - $(2 \times 50\% = 100\%)$
 - High pulse is UN-DAMPED
 - Low pulse is DAMPED



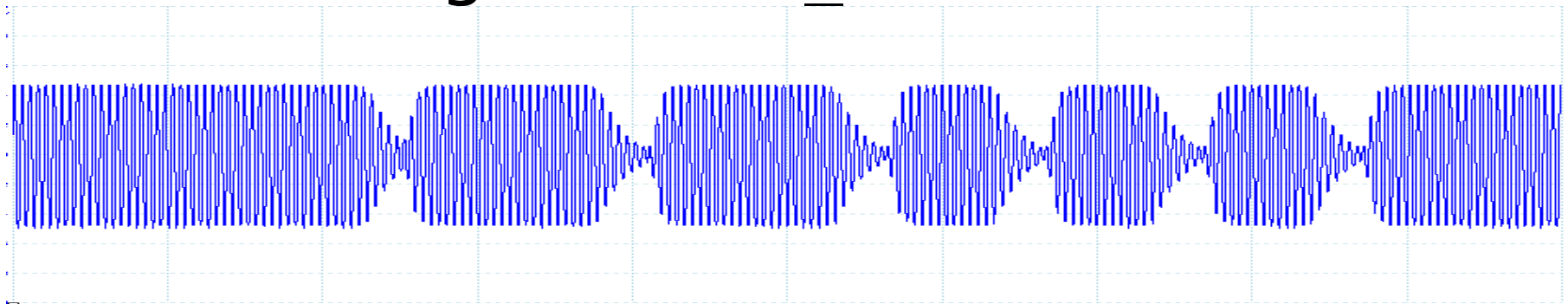
RFID Basics

- ASK / PSK – Phase Shift Keying
 - 1 bit per period
 - Phase change = value change
 - 011010100011111001110001000101100001111010011100101101100001



RFID Basics

- Modulation schemes
 - PWM – Pulse Width Modulation
 - '1' is a long pulse, '0' a short
 - This message is '11000'
 - Hitag2 'START_AUTH'



Passive TAGs

- One-way communication:
 - TAG → READER
 - Fixed ID
 - Plaintext
 - Even 'encrypted' is fixed – i.e. no session key
- About as secure as a barcode!
 - EM4102
 - HID Prox (plaintext content)
 - Indala (encrypted content)

Active TAGs

- Two-way communication:
 - READER → TAG & TAG → READER
 - Fixed or Random ID
 - May be encrypted
 - Session key
 - Two-Way Authentication
 - As secure as underlying crypto
 - Hitag2 (broken)
 - DESFire (DES, 3DES, AES)

RFIDler LF (125/134 KHz)

- Very low cost
 - Standard: Full device with processor
 - USB / TTL CLI / API & GPIO
 - £30.00
 - Lite: RFID Coil & ASK mod/demod only
 - GPIO
 - £20
 - Kickstarter project

RFIDler LF (125/134 KHz)

- Utilise ANY modulation scheme, including bi-directional protocols
- Write data to tag
- Read data from tag
- Emulate tag
- Sniff conversations between external reader & tag
- Provide raw as well as decoded data
- Built-in antenna
- External antenna connection
- USB power and user interface
- TTL interface
- GPIO interface
- JTAG interface for programming
- USB Bootloader for easy firmware updating

RFIDler LF (125/134 KHz)

- EM4102 / Unique
- Hitag 1/2/S
- FDX-B (ISO 11784/5 Animal Standard)
- Q5
- T55xx
- Indala
- Noralsy
- HID Prox
- NXP PCF7931
- Texas Instruments
- VeriChip
- FlexPass

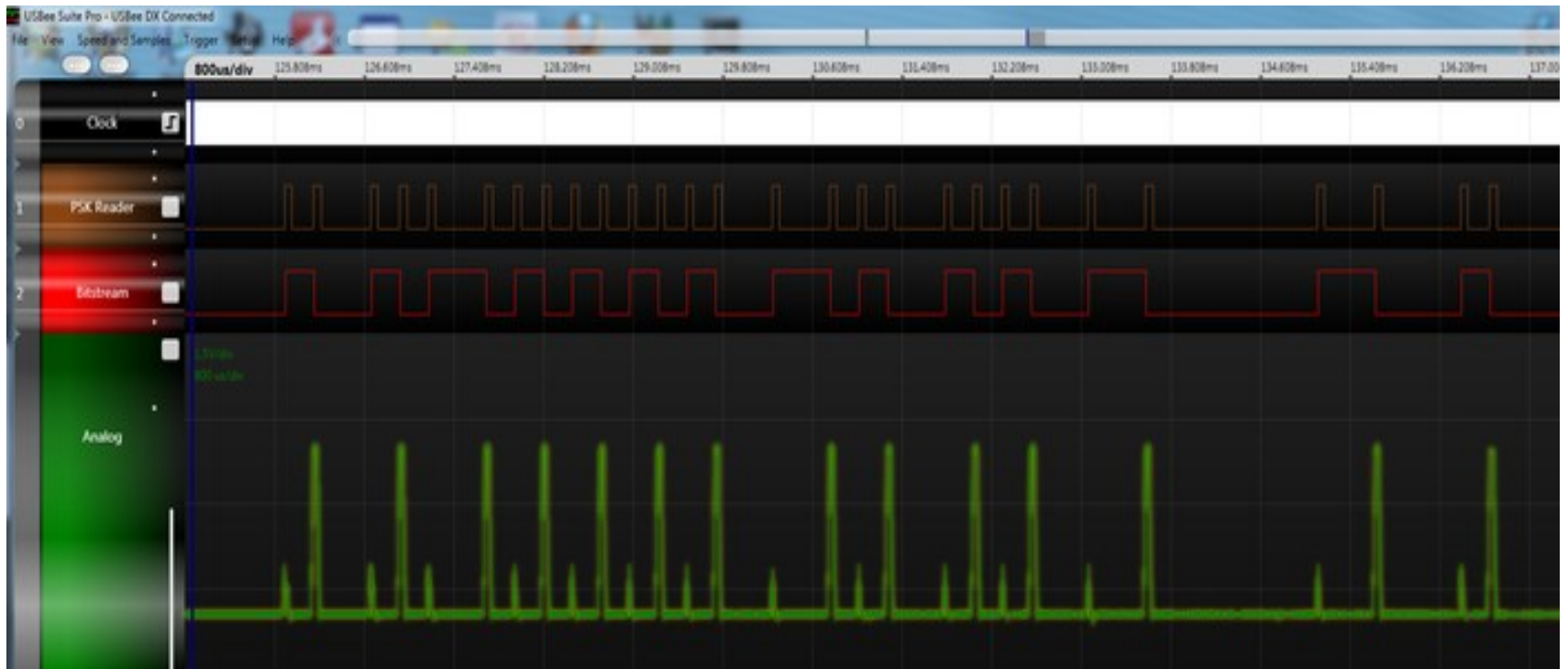
RFIDler LF (125/134 KHz)

How SD is it?

- Hardware Modulate / Demodulate:
 - ASK
- Software Modulate / Demodulate:
 - CARRIER
 - FSK / PSK
 - Manchester / BiPhase
 - PWM

RFIDler LF (125/134 KHz)

Reading PSK



RFIDler LF (125/134 KHz)

Emulation / Commands

- Measure in Field Clocks
 - $1 \text{ second} / \text{Frequency} == 1 \text{ Field Clock}$
 - e.g. $1 / 125\text{KHz} == 8 \text{ uS}$
- Baseband timings from datasheets

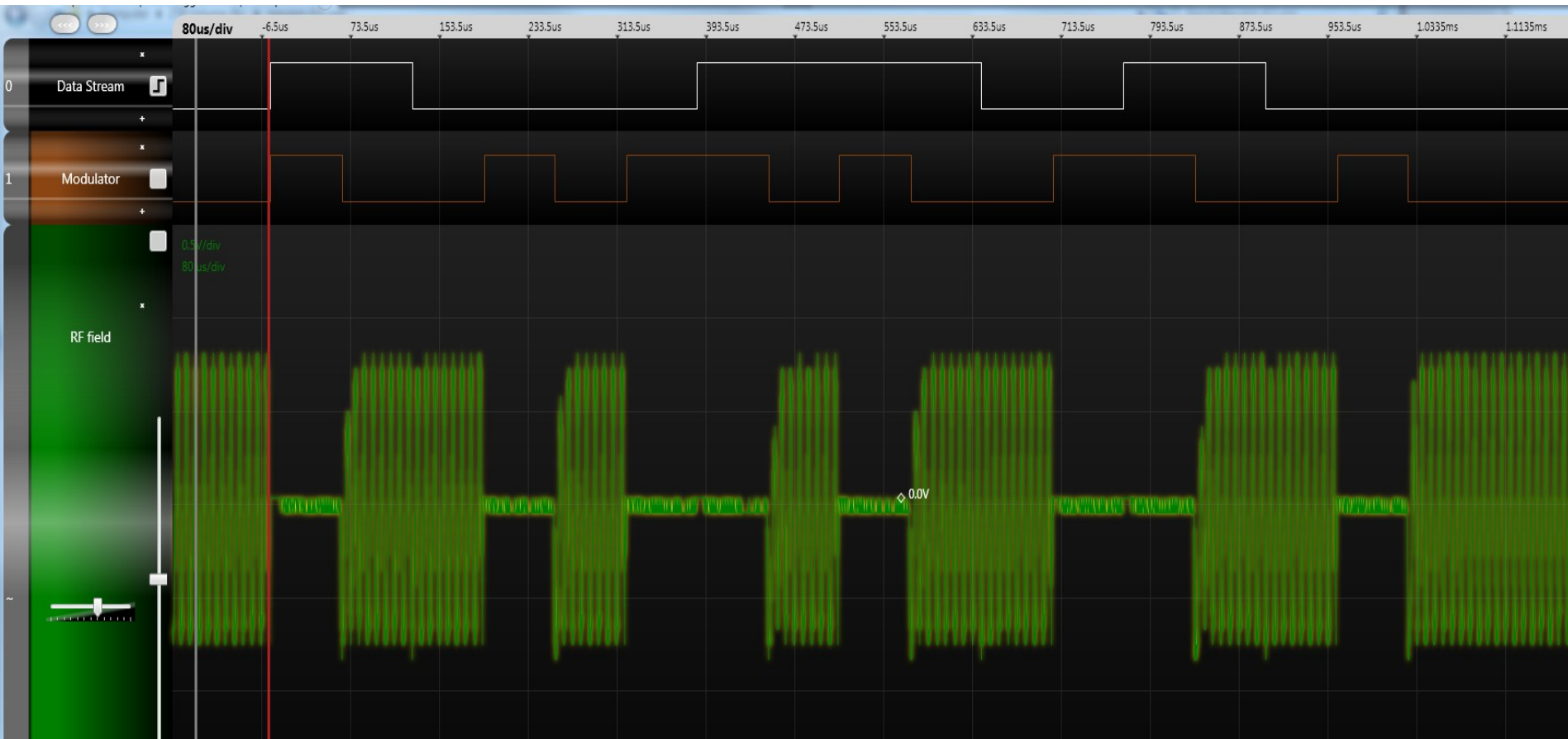
RFIDler LF (125/134 KHz)

Emulating ASK



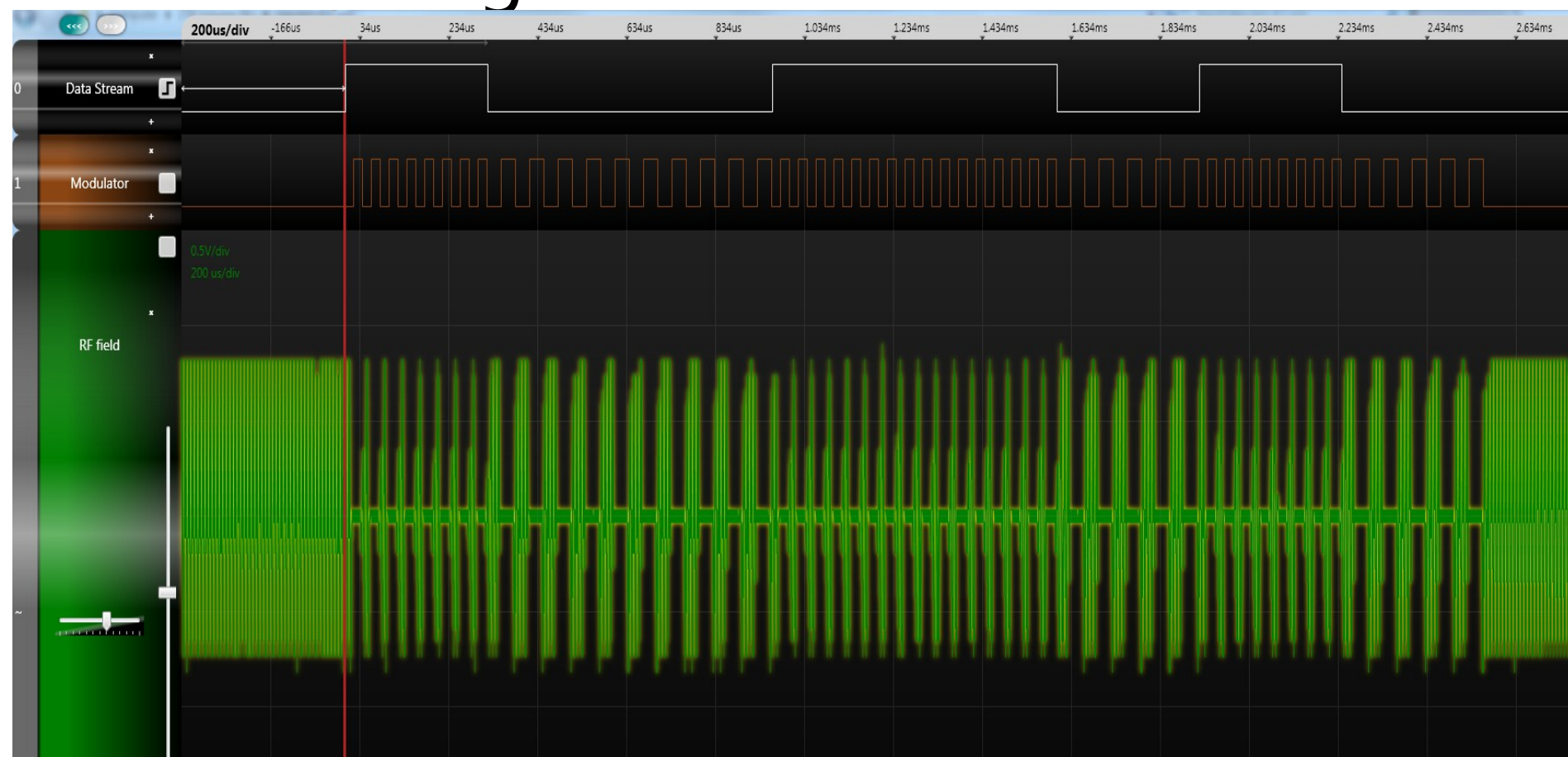
RFIDler LF (125/134 KHz)

Emulating Manchester



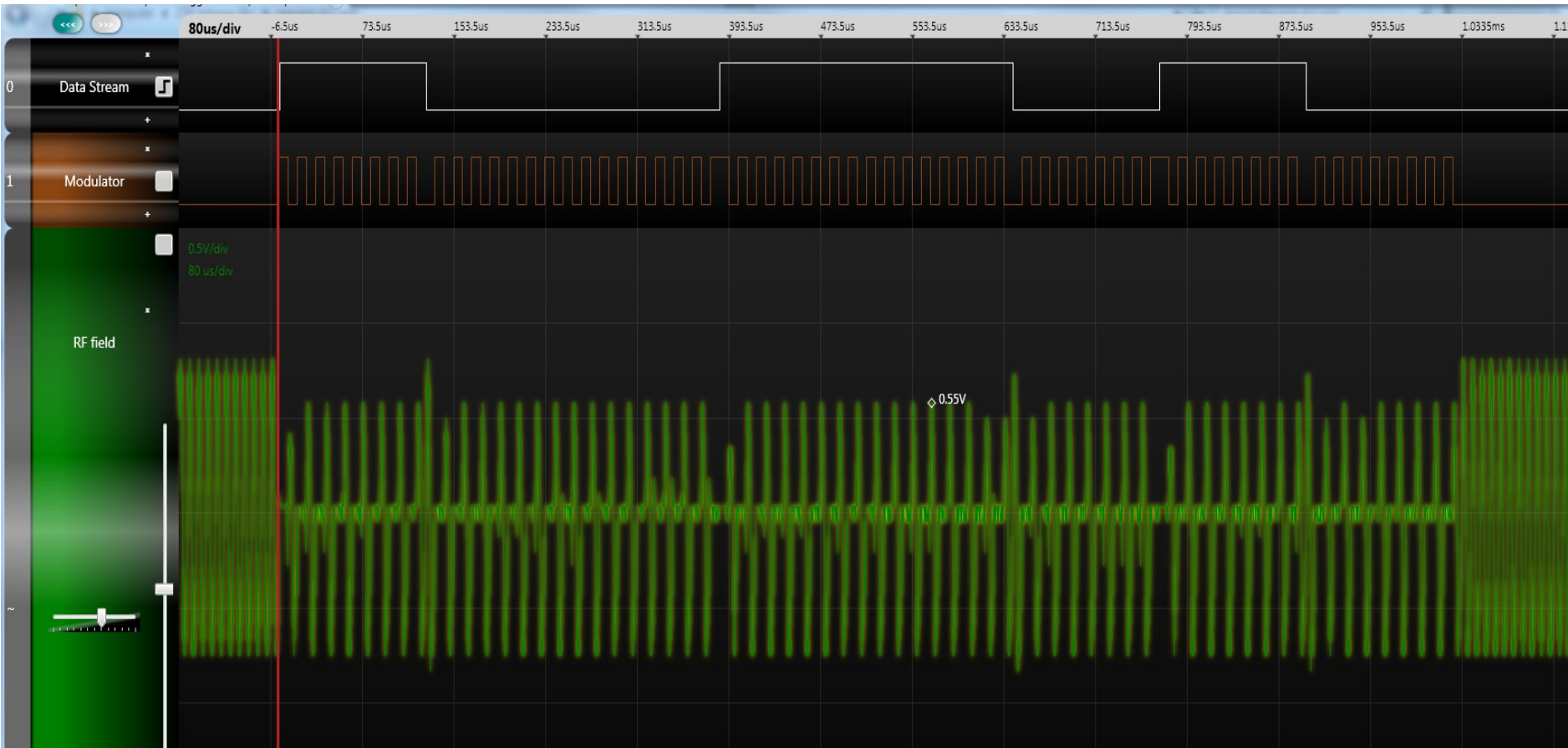
RFIDler LF (125/134 KHz)

Emulating FSK



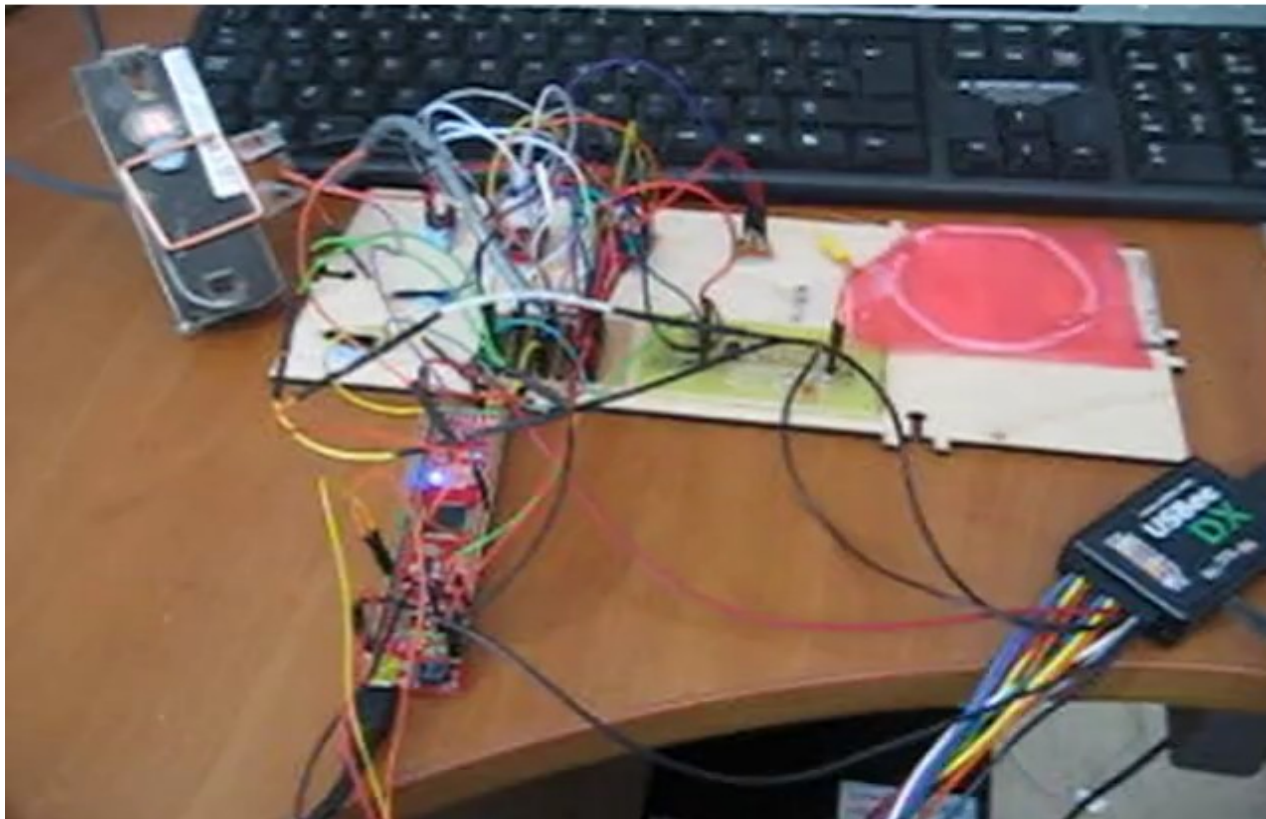
RFIDler LF (125/134 KHz)

Emulating PSK



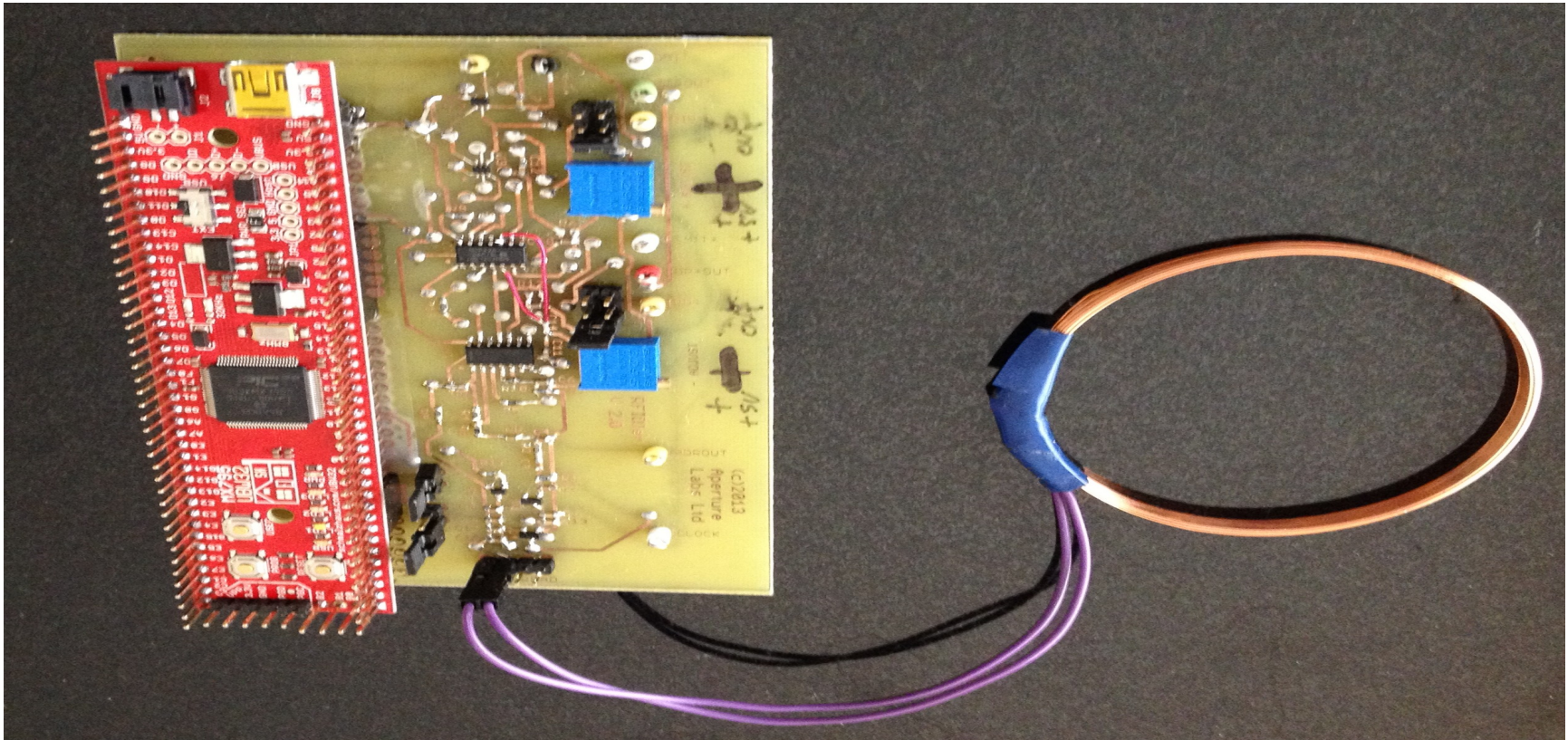
RFIDler LF (125/134 KHz)

Prototype 1

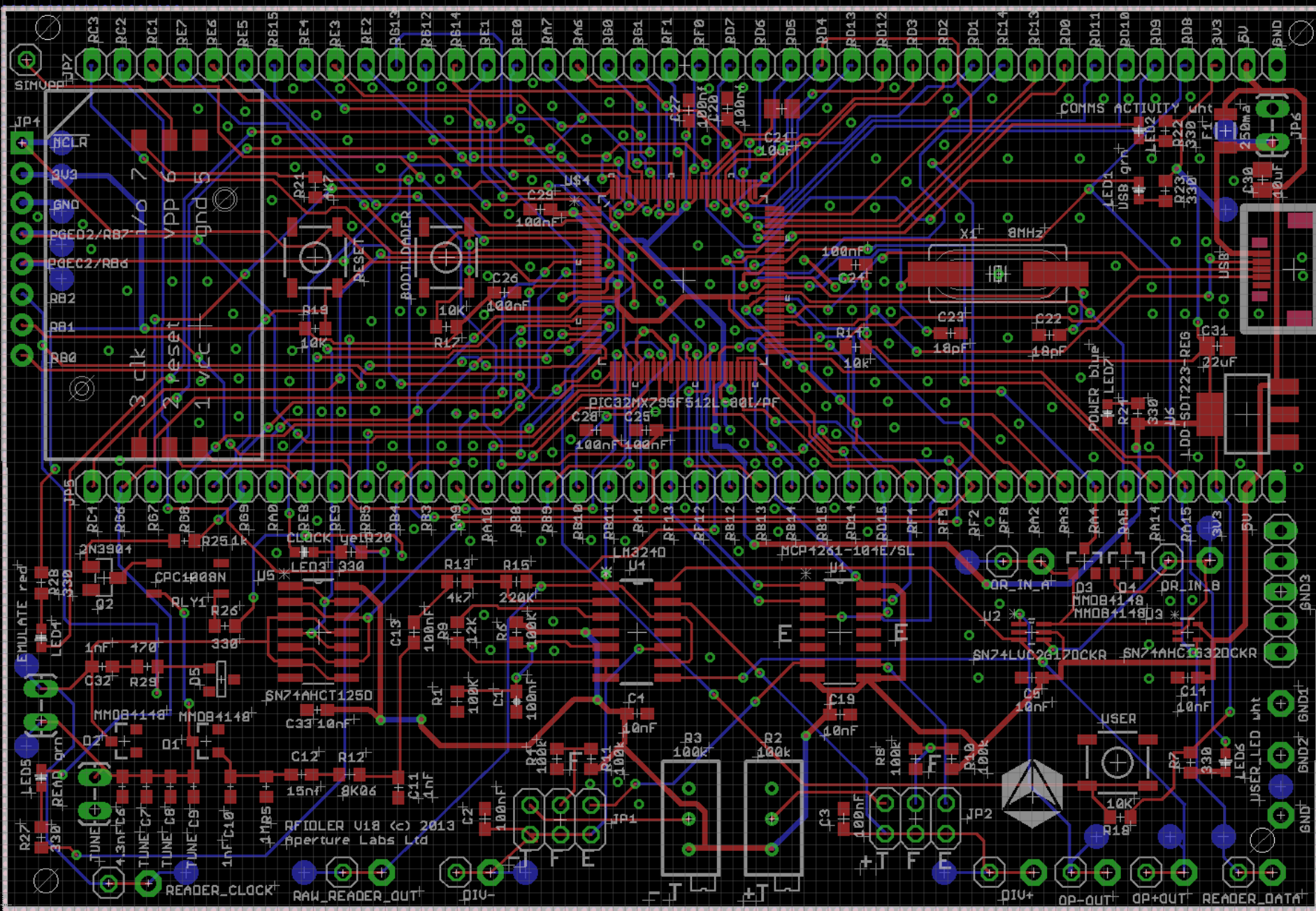


RFIDler LF (125/134 KHz)

Prototype 2

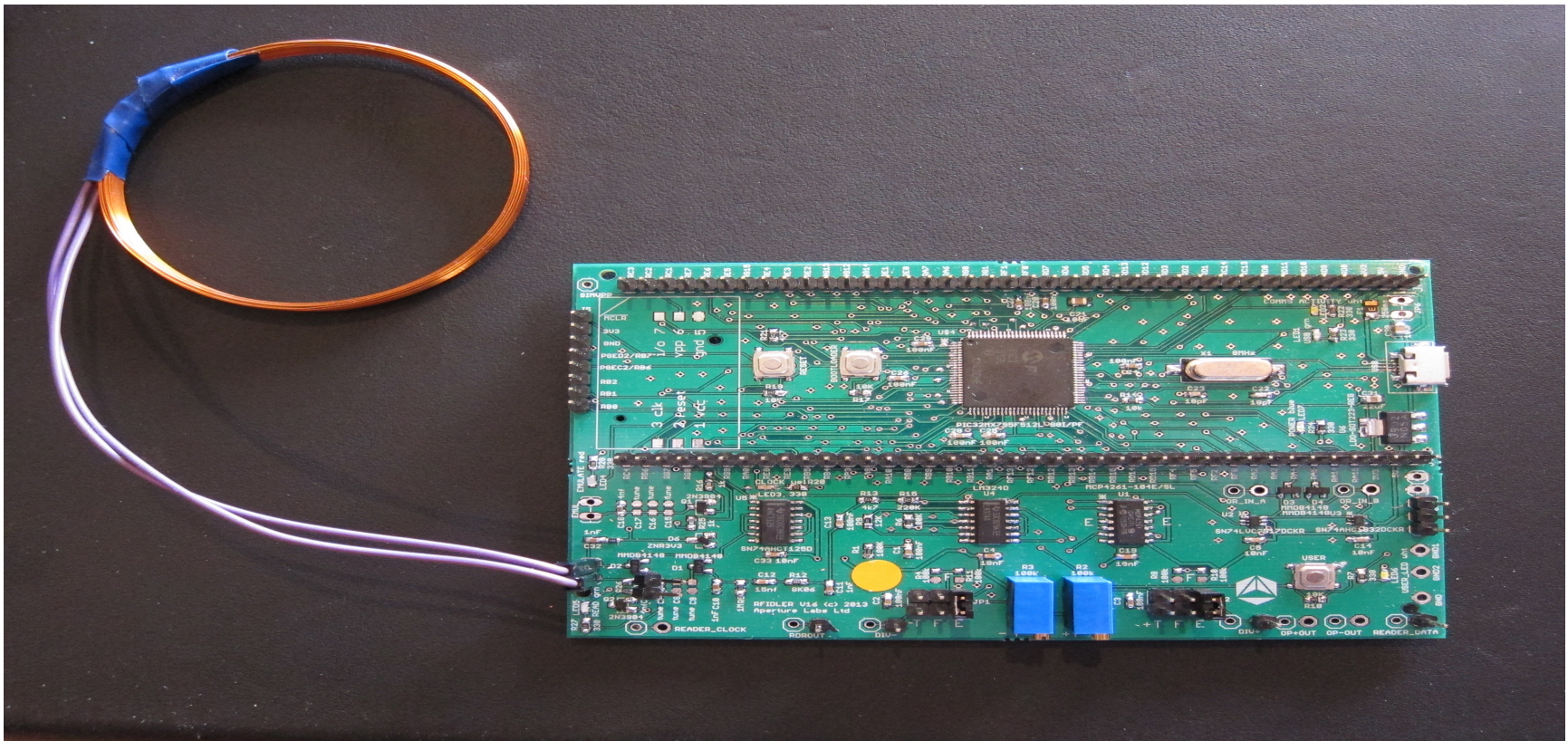






RFIDler LF (125/134 KHz)

Prototype ... 20(!)



RFIDler LF (125/134 KHz)

DEMO

RFIDler LF (125/134 KHz)

So what?

It's all been done before...

RFIDler LF (125/134 KHz)

AWID



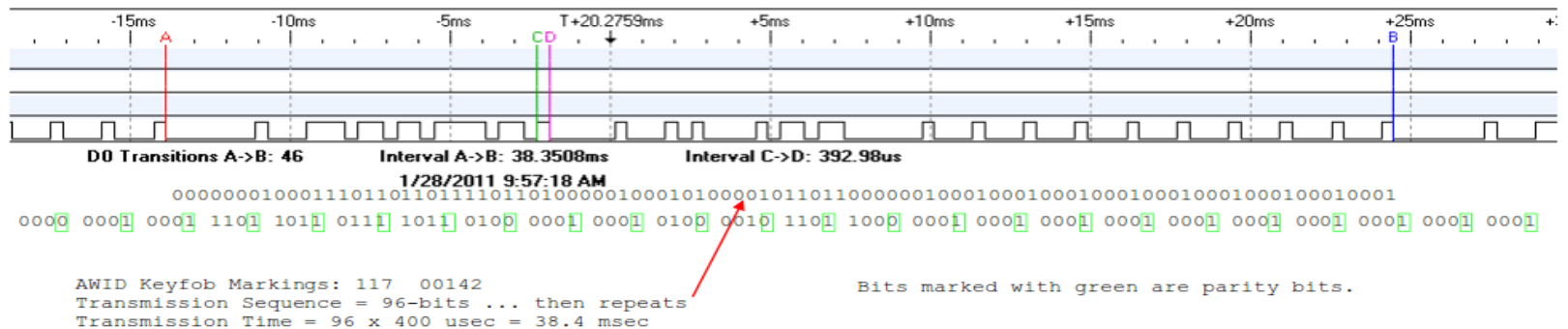
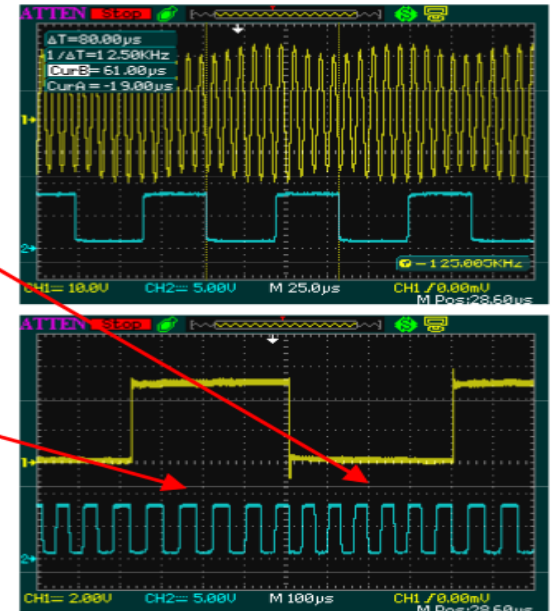
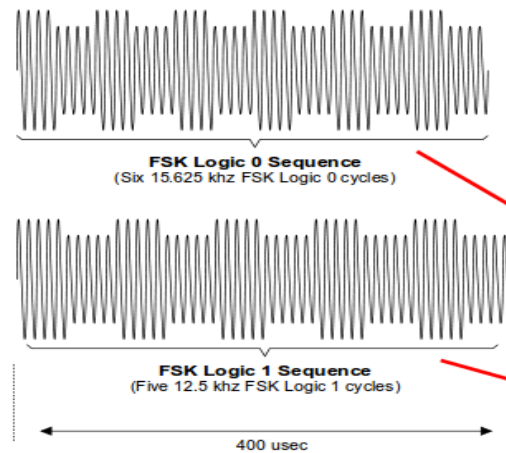
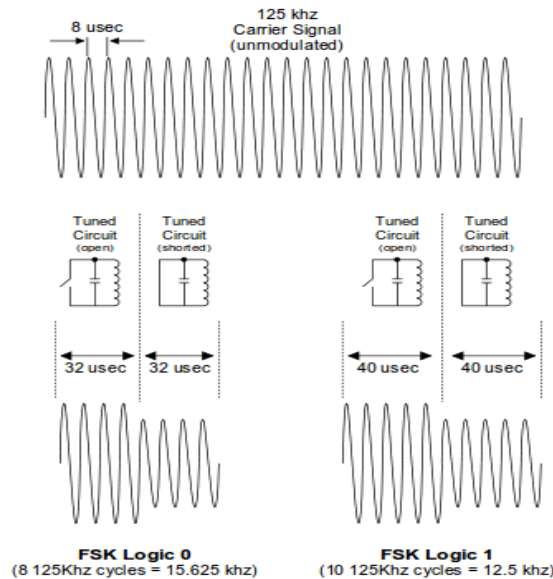
RFIDler LF (125/134 KHz)

AWID



http://www.proxclone.com/pdfs/AWID_FSK_Format.pdf

RFIDler LF (125/134 KHz)



WIRING Show wiring diagram (alias: WIRES PINS)

Note: '*' prepended to main prompt indicates unsaved config changes

RFIDler> set tag awid26

OK

*AWID26> encode 12300666 q5

OK

*AWID26> vtag

Type: Q5

Emulating: AWID26

Raw UID: 011D87DD1247411111111111

UID: 12300666

Data:

0: E6018056

1: 011D87DD

2: 12474111

3: 11111111

*AWID26> clone

1: 011D87DD

2: 12474111

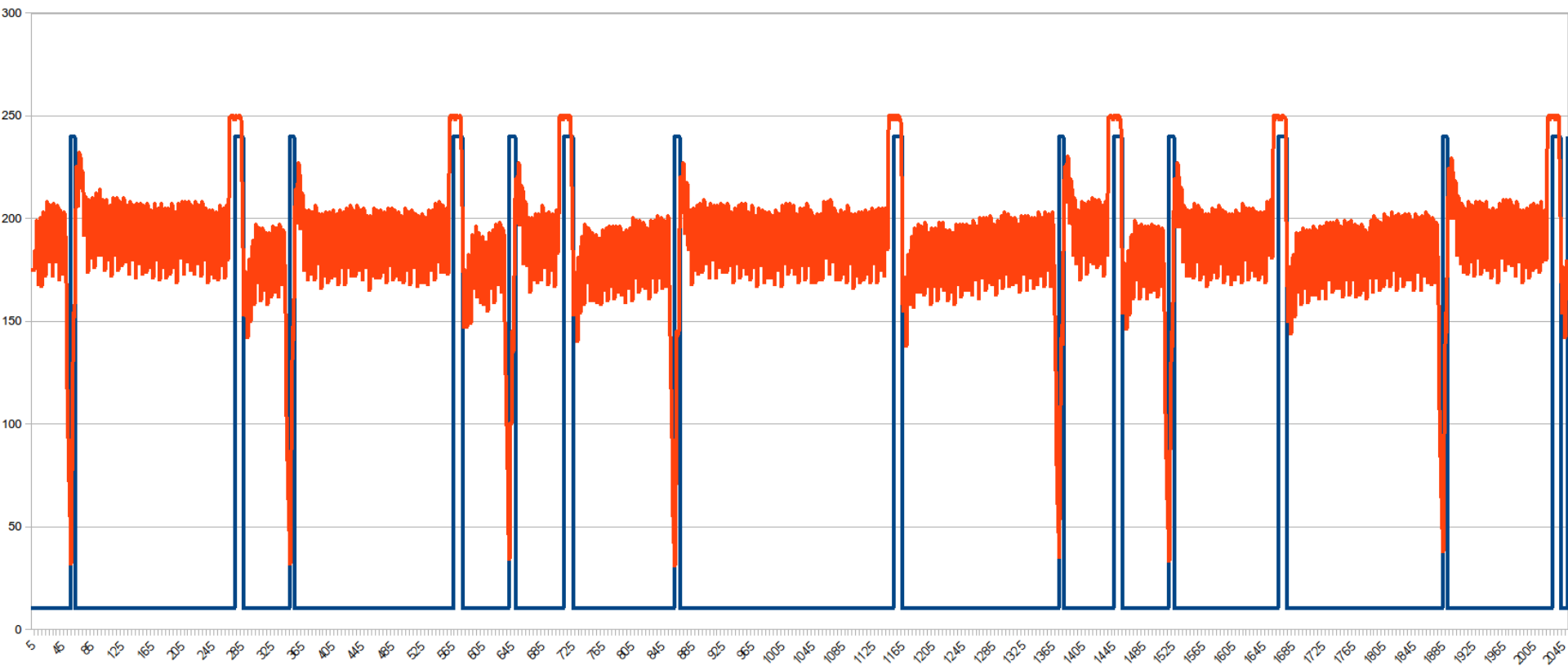
3: 11111111

0: E6018056

OK

*AWID26>

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.6.2 | VT102 | Offline



RFIDler LF (125/134 KHz)

Questions?

<https://github.com/ApertureLabsLtd/RFIDler>

<http://www.kickstarter.com/projects/1708444109/rfidler-a-software-defined-rfid-reader-writer-emul>